Methodologies for Integration of PHM Systems with Maintenance Data

Fatih Camci
Impact Technologies, LLC
200 Canal View Blvd.
Rochester NY, 14623
fatih.camci@impact-tek.com
(585)424-1990

G. Scott Valentine
Impact Technologies, LLC
200 Canal View Blvd.
Rochester NY, 14623
scott.valentine@impact-tek.com
(585)424-1990

Kelly Navarra
Wright Patterson AFB
2230 10th Street, OH 45433
Kelly.navarra@wpafb.af.mil
(937)255-1303

Abstract—The Automatic logistics program in the Air Force seeks to reduce development, production, and ownership costs for the next generation fighter aircraft by increasing system reliability while reducing maintenance requirements. A large number of technologies are becoming available within the Prognostics and Health Management (PHM) community that will lead to reduced cost and increased availability. [1 2]

The challenge is to develop advanced technology to integrate available PHM information from a variety of different sources into the maintenance and logistics infrastructure. PHM and maintenance/logistics systems must be thoroughly examined and tightly integrated in order to perform maintenance actions in the most efficient way to reduce ownership cost and increase availability. This paper presents multi-agent technology that integrates maintenance and PHM data to provide more effective maintenance identification and scheduling.

The proposed methodologies will enable the maintenance and logistics infrastructure to fully benefit from newly developed PHM systems. Additionally, the PHM systems update themselves based on feedback obtained from the maintenance systems. The integration will utilize intelligent software agent technology in order to develop such solutions within open, highly dynamic, uncertain and complex environments with data distributed over a network. This provides benefits such as reusability, scalability, and continuous improvement with dynamically evolving ability.

TABLE OF CONTENTS

1. INTRODUCTION

A Prognostics Health Management system is one of key components of the JSF Autonomic Logistics (AL) system architecture [1][2]. A large number of technologies are becoming available within the PHM community that enables improved fault detection, advanced diagnostics, and prognostics in aerospace systems [3-9]. Advances in sensor, health assessment, diagnostics, prognostics, and decision support technologies have produced a wide variety of potential maintenance solutions. The challenge is to develop advanced technology to integrate this available PHM information from a variety of different sources into the maintenance and logistics infrastructure. Moreover, it is desired to specify and develop such solutions within open, highly dynamic, uncertain and complex environments which have data distributed over a network. This provides benefits such as reusability, scalability, and continuous improvement with dynamically evolving ability. These benefits are widely sought after by the Air Force and other DoD program offices.

The USAF automatic logistics program seeks to reduce development, production, and ownership costs for the next generation fighter aircraft by increasing system reliability while reducing maintenance requirements. PHM enables maintenance to be planned on the basis of actual component or system health state. This represents a key component within the autonomic logistics system architecture. Hence, both PHM and maintenance/logistics systems must be thoroughly examined and tightly integrated in order to perform maintenance actions in the most efficient way that will lead to reduced ownership cost and increased availability. This article presents an intelligent software agents tool to analyze, negotiate and optimize decisions regarding database adaptation, maintenance, and logistics actions in a self-learning environment.

1———————————

[1] 1-4244-0525-4/07/$20.00 ©2007 IEEE.
[2] IEEEAC paper #1191, Version 1, Updated September 20, 2006

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **SEP 2006**	2. REPORT TYPE	3. DATES COVERED **00-00-2006 to 00-00-2006**

4. TITLE AND SUBTITLE **Methodologies for Integration of PHM Systems with Maintenance Data**	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Impact Technologies LLC,200 Canal View Blvd,Rochester,NY,14618**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT
see report

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**9**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

2. SYSTEM OVERVIEW

The system overall layout of the proposed PHM and Maintenance data integration tool is displayed in Figure 1. PHM data consist of Remaining Useful Life (RUL) (i.e., prognostic) and failure mode (i.e., diagnostic) as represented on the left in the figure. Maintenance data include resources (parts, personnel, material, and tools etc.) required for maintenance actions, available resources in the inventory, lead time for resources when ordered, etc. The tool also analyzes the planned mission information in order to obtain a more accurate RUL since mission profiles affect the rate of in equipment health degradation.

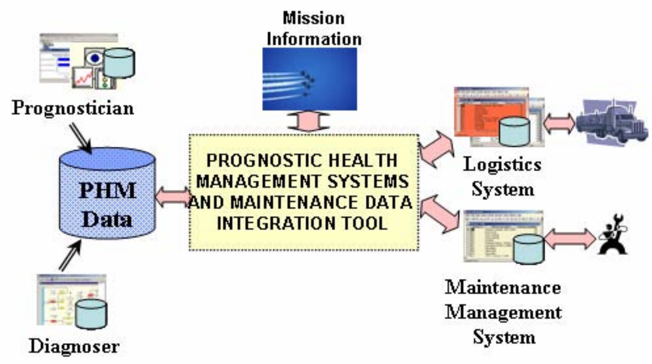


Figure 1: System Overall Layout

3. MODELING

The integration tool consists of seven intelligent agents, each of which has separate goals and communicates with each other in order to achieve the ultimate goal of the tool. Figure 2 illustrates the modeling layout. Agents are discussed in detail in the following sections.

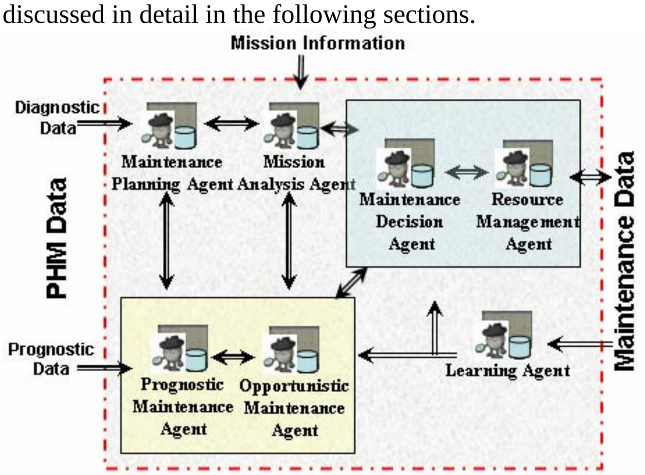


Figure 2: Agent Modeling

Maintenance Planning Agent

The Maintenance Planning Agent (MPA) identifies the maintenance task ranking provided the failure mode from PHM algorithms. A standard failure mode and criticality analysis of the system provides the basic information required by the algorithm.

MPA performs maintenance task identification, which is the recommendation of a corrective action based on information obtained from system and PHM data. Maintenance tasks associated with each failure mode are ranked based on user-defined weighting factors, e.g. task effectiveness, cost, downtime, etc.

The problem is a function of maintenance effectiveness for the failure mode, maintenance downtime and cost, as defined below. The goal of maintenance task identification is to select the optimal maintenance task based on minimum downtime and cost. For example, assuming that cost and downtime carry equal weight, the optimal maintenance task is associated to the point with the minimum distance to the virtual solution as defined by the Euclidian distance (shown in Figure 3).

$$R_i = 1 - \sum_{k=1}^{n} w_k \left(x_{ki} - \min_j x_{kj}\right)^2$$

Where:

R_i	=	Rank for i^{th} Maintenance Task
n	=	Number of variables (Downtime,Cost..)
$x_{k,i}$	=	Index to value of k^{th} variable for i^{th} maintenance
w_k	=	Weighting Factor for k^{th} variable.
$\min(x_{k,i})$	=	Minimum value from variable vector.

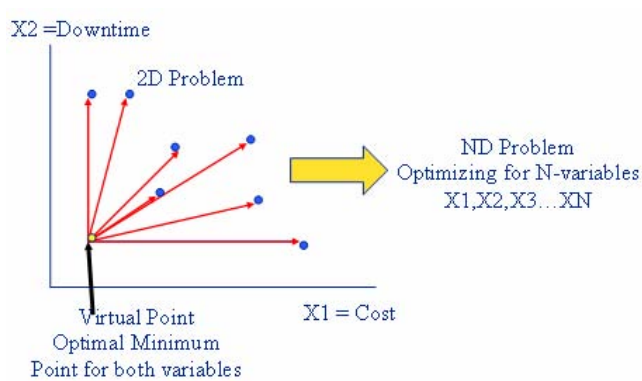


Figure 3: Multi-objective Optimization

Mission Planning Agent

Any Remaining Useful Life (RUL) estimation must consider the future mission plan of an aircraft. The effect of mission types on the component degradation varies depending on the difficulty of the mission. Equipment may degrade more during some missions compared to the others such as refueling compared to air combat. Figure 4 illustrates three mission types with different mission length

scheduled for the given period. As seen from the graph, mission 1 (M1) causes the most degradation even though it lasts the least time. Mission 3 (M3) causes less degradation even if it is closer to the failure. The mission Planning Agent (MisPA) will model the relationship between RUL and mission planned.

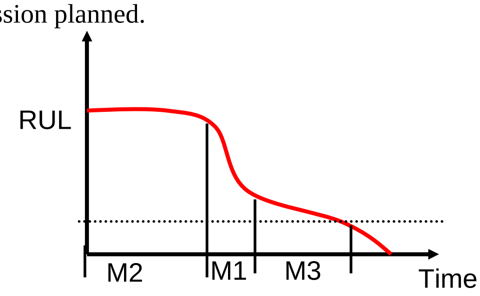


Figure 4: RUL with mission

MisPA integrates prognostic information (e.g., RUL) and planned mission schedule in order to more accurately estimate the time of the possible failure. The agent input-output schema is illustrated in Figure 5.

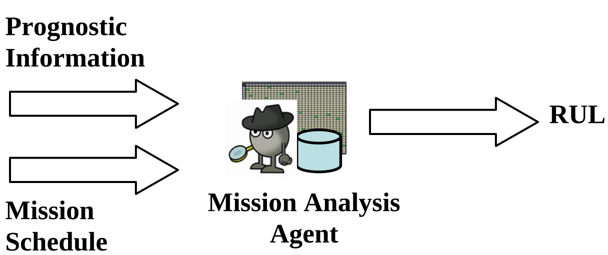


Figure 5: Mission Planning Agent Input-Output Schema

MisPA calculates what the RUL will be after a given mission is completed as follows:

$$RUL_{norm,j} = RUL_j - \sum_{i=1}^{N} l_{i,j} \times c_i$$

RUL_j	RUL for aircraft *j received from PHM data*
$RUL_{norm,j}$	Normalized RUL based on planned missions for aircraft *j*, what the RUL would be after the mission
$l_{i,j}$	Length of the mission *i* for aircraft *j*
c_i	Depreciation coefficient of mission *i*

The depreciation coefficient, c_i is the key parameter to differentiate the depreciation effects of missions. There exist two possible ways of obtaining this parameter: 1) Historical data analysis 2) Mission related-material analysis. Mission material analysis obtained through Physic based models will lead to more accurate answers. To the best of our knowledge, it is feasible to perform this analysis using stress factors collected off the airframe and collecting manufacturing information from OEMs. MisPA can be applied to prognostic information, which does not already include mission information. In rare cases, prognostic methods may consider the scheduled mission of the aircraft. Implementation of MisPA on such cases may mislead the result, since mission information would be considered twice.

Prognostic Maintenance Agent

Prognostic Maintenance Agent (PMA) receives the modified RUL information from MisPA and recommends times of the maintenance actions for the given equipment based on two thresholds: T :Required Maintenance Threshold, and τ :Opportunistic Maintenance Threshold. The estimated RUL values at the end of each scheduled mission are compared with thresholds in order to decide a possible maintenance. The time that RUL becomes less than the first threshold (T) is identified as the required maintenance time for that component. If there exists a mission planned for this time, the maintenance time is moved to the closest time before the mission starts. No maintenance is scheduled for the period that RUL value is greater than τ . Opportunistic maintenance options will be considered for the period where RUL is between these two thresholds: $\tau > RUL > T$. The components are analyzed independently in PMA. This information is sent to Opportunistic Maintenance Agent (OMA) for further analysis. Figure 6 illustrates RUL threshold setting.

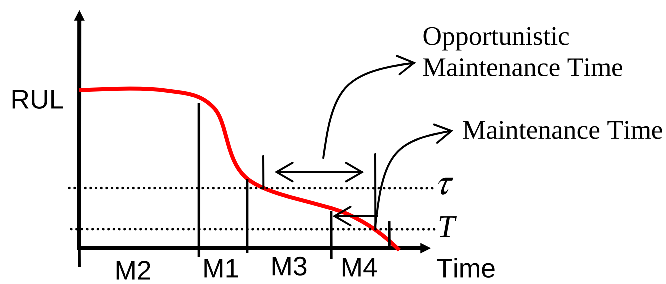


Figure 6: RUL Threshold Set

The threshold values are identified by optimizing the component availability and ownership cost. These parameters will also be updated by learning agent as feedback from the maintenance operator becomes available.

Opportunistic Maintenance Agent

Opportunistic Maintenance actions are conducted to improve a system's performance and or availability when it is convenient due to related circumstances. The Opportunistic Maintenance Agent (OMA) receives the recommended maintenance times from PMA along with possible opportunistic maintenance time periods as input. OMA identifies the opportunistic maintenance actions that minimizes the cost and maximizes the equipment availability. In other words, OMA performs the system analysis using results received from PMA that are obtained from independent individual component analysis. Figure 7

illustrates three components with the same mission schedule. As annotated by "P", component 2 and 3 are recommended for maintenance at time t_2 and t_1, respectively. OMA analyzes different maintenance scheduling options such as maintaining component 1 with component 2 or 3. It also analyzes performing three maintenance actions at the same time.

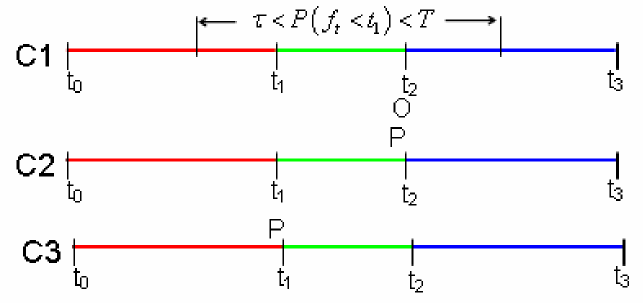


Figure 7: Maintenance Schedule for 3 components

OMA searches for different maintenance options that will lead to less cost and more equipment availability based on the following equations:

$$E[Cost] = E[Main'tCost] + E[FailureCost]$$

$$E[FailureCost] = C_f \int_{t_0}^{t_1} P(x)dx$$

$$E[Main'tCost] = C_M \sum^{t_1} M_t$$

$$M_t = \begin{cases} 1 & \text{If maintenance is scheduled at time t} \\ 0 & \text{Otherwise} \end{cases}$$

$$E[Cost] = C_M \sum_{i=t_0}^{t_1} M_t + C_f \int_{t_0}^{t_1} P(x)dx$$

Resource Management Agent

The applicability of the scheduled maintenance actions is tied to the availability of tools, parts and personnel. The Resource Management Agent (RMA) retrieves the resource information from the maintenance database and checks for availability. It sends a confirmation message to the agent (MPA or PMA), if all resources are available for the maintenance. If not available, it checks for the lead time for these items to be delivered to the facility if ordered and sends a message to the agent (PMA or MPA) about different feasible alternatives for the maintenance. Figure 8 illustrates the process flow for RMA.

Resource Planning Agent also performs resource allocation that provides the effective usage of limited resources. RPA identifies the most important maintenance actions to perform if resources are limited and needed by several maintenance actions. Figure 9 illustrates the inventory with the given maintenance schedule. As seen from the figure, RMA gives an order for missing parts if the parts can be delivered on time. If not, it checks for different maintenance options with given parts by removing the least important maintenance action based on PHM data.

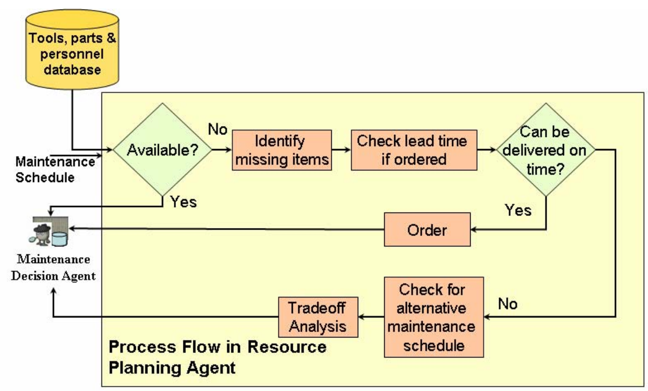


Figure 8: Resource Management Process Flow

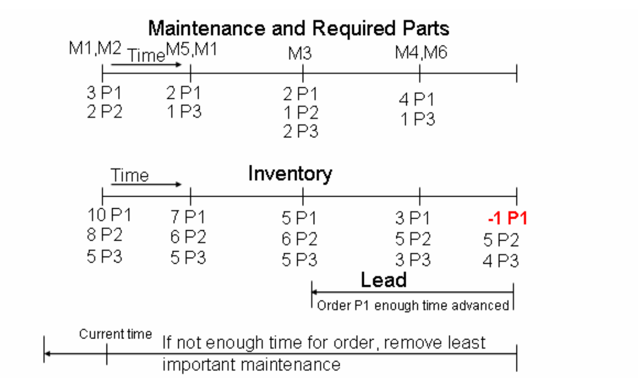


Figure 9: Inventory Management and Resource Allocation

Learning Agent:

As experience is accumulated, some of the parameters within the model can be learned by analyzing the feedback from the maintainer. The parameters to be learned are opportunistic maintenance threshold (τ), required maintenance threshold (T), resource lead time, and maintenance effectiveness. In order to develop learning for these parameters, the first step is to identify the questions to be asked to the user. The following section summarizes the feedback analysis for these parameters.

Opportunistic Maintenance and Required Maintenance Thresholds (τ, T):

This parameter is a threshold for RUL value, which is used

to identify the opportunistic and required maintenance time period for the component. If the RUL of a component is greater than the opportunistic maintenance threshold, there is no need for any maintenance; if it is less than opportunistic maintenance threshold, and greater than required maintenance threshold, an opportunistic maintenance is valuable whenever it is convenient within this time frame. Otherwise, maintenance action is required as soon as possible.

Asking the operator the question “Was the maintenance necessary?” will help update these parameters. For example, an answer of “No, it was not necessary” means the thresholds were too high and will lead us to reduce the thresholds.

Resource Lead Time:

In the case of a shortage resources will need to be ordered. The expected lead times are processed within the model in order to identify the feasibility of the maintenance schedule. If the lead times after ordering the resources turn out to be different from values in the database, they will be updated with the new numbers.

Maintenance Effectiveness:

Maintenance effectiveness is a value that ties maintenance action to be performed to the failure mode. This parameter is important in identifying the correct maintenance actions. If a maintenance action does not work for a specific failure mode, the effectiveness value between this maintenance action and failure mode should be weakened. Oppositely, if a maintenance action corrects the problem, the effectiveness value should be increased. The operator will answer the question “Did the maintenance work” as “yes” or “no”.

The parameters mentioned above will be updated according to the following equation:

$$p = p + \alpha\beta$$

$$\beta = \begin{cases} 1 & \text{if answer yes} \\ -1 & \text{if answer no} \end{cases}$$

α : Learning rate

p : Parameter to be learned

Decision Management Agent

Decision Management Agent (DMA) can be defined as the manager of the software tool. DMA takes the maintenance times and associated ranked maintenance actions and availability of resources and then reports the current status to the user. The user authorizes MPA to send maintenance orders to the maintenance operator. MPA also has ability to learn so that it can send maintenance orders without human authorization if it has enough confidence. Historical cases in conjunction with their success rates are utilized for learning.

4. MULTI AGENT PROGRAMMING

Multi-agent technology is perfectly compatible with the adaptive maintenance/logistics and PHM knowledgebase infrastructure. Agent-based technologies are appropriate in applications with some or all of the characteristics summarized below (the first three are highly relevant to the Autonomic Logistics Program) [10]:

- The environment is open, highly dynamic, uncertain, or complex
- Distributed data, control or expertise are hallmarks of the system
- Agents are a natural metaphor to model interacting entities collaborating (or competing) to solve a complex problem or achieve a goal
- Use of legacy systems requiring “wrapping” for compatibility is mandated

Java Agent DEvelopment Framework (JADE) is both an open-source software framework to write agent applications conforming to the Foundation for Intelligent Physical Agents (FIPA) specifications and a runtime execution environment (Container) for the agents developed using the JADE Application Programming Interface (API). FIPA defines a reference model of an agent platform and a set of services that should be provided. Adherence to FIPA standards ensures that JADE agents can communicate with other agents in compliance with these specifications [13].

5. AGENT COMMUNICATION

This section summarizes the communication between agents. Agents communicate by sending and receiving encoded messages. The messages are sent from RPA to MPA and MPA separates it into meaningful parts as shown below.

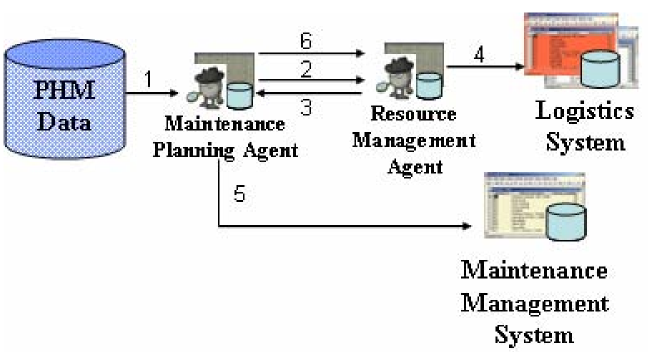
Message Content:
0,156KMRN-1-1-o-3-nnMnnnnnn, nnnnnnMnn, nnMnnnnnn, nnMnnnnnn, nnMnnnnnn- nnooommmm, nnnnoommm, nnmmmmmmm, oooommmmm, nnoommmmm,

Table 1: Message Encoding

Message Code	**Meaning**
0	Infeasible schedule
156KMRN	Maintenance Action Code that cause the infeasibility
1-1-o-3	**1st Character:** Number of resource that the maintenance action requires

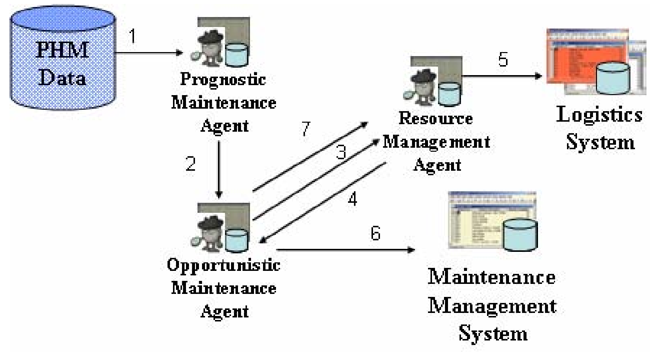
1-**1**-o-3	**2nd Character:** Number of missing resource in the inventory
1-1-**o**-3	**3rd Character:**: Maintenance type (opportunistic maintenance)
1-1-o-**3**	**4th Character:** There are 3 more time units until the time that the component needs required maintenance
nnMnnnnnn,….	Maintenance Schedule for components for given time period (n: no Maintenance, M: Maintenance)
nnooommmm,….	Maintenance pre-scheduling based on RUL analysis (n: no maintenance, o: opportunistic maintenance, m: required maintenance)

Figure 10 and Figure 11 illustrate the communication between MPA & RPA and PMA, OMA & RPA, respectively. Figure 12 illustrates the communication between Learning Agent and other agents in order to learn the model parameters.



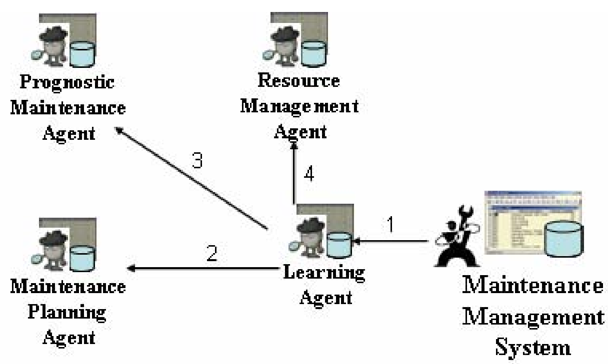
1: New information triggers MPA
2: Maintenance action recommended
3: Maintenance rejected or accepted
4: Order missing parts
5: Give maintenance orders
6: Recommend an alternative if the previous is rejected

Figure 10: MPA – RPA Communication



1: New information triggers PMA
2: Initial maintenance schedule is proposed
3: Maintenance schedule with opportunistic is proposed
4: Send resource availability
5: Order missing parts
6: Give maintenance orders
7: Recommend an alternative if the previous is rejected

Figure 11: PMA – OMA – RPA Communication



1: Feedback triggers LA
2: LA updates the maintenance effectiveness used by MPA
3: LA updates the threshold values used by PMA
4: LA updates the resource lead time used by RMA

Figure 12: Learning Agent Communication

6. SOFTWARE IMPLEMENTATION

This section discusses the implementation of the intelligent agent software tool. When the program is executed, the agents are initiated and wait for a trigger, which is diagnostic or prognostic information being saved to the database. Figure 13 displays the five agents that wait for a trigger.

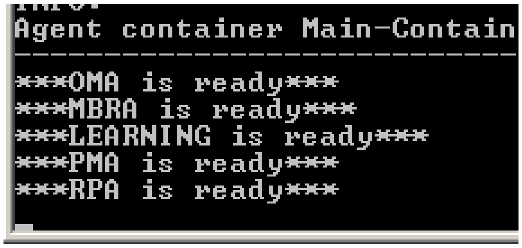


Figure 13: Agents Ready Status

Implementation of Maintenance Planning Agent

The maintenance planning agent regularly checks the availability of new diagnostic information. When MPA detects new diagnostic information, it analyzes this information and identifies the best maintenance action for the given diagnostic information. After the identification of the maintenance action, it sends a message to resource planning agent (RPA) and waits for RPA's response about the feasibility of the recommended maintenance action. If approved by RPA, MPA sends the maintenance order; if not, it recommends an alternative maintenance action. Figure 14 displays the communication between MPA and RPA.

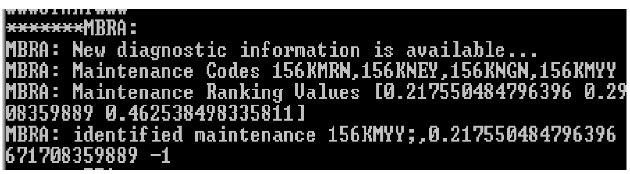


Figure 14: Maintenance Planning Agent

Implementation of Prognostic Maintenance Agent

Prognostic Maintenance Agent (PMA) regularly checks for new available prognostic information. If new prognostic information is available, it classifies the given time frame for each component as "No maintenance", "opportunistic maintenance", or "time based maintenance" utilizing prognostic information. This identification process is based on expected risk reduction criteria obtained from RUL values. In PMA, all components are treated independently. Thus, the advantage of performing maintenance for multiple components at the same time is not considered. PMA sends the independently analyzed component time frames to OMA for opportunistic maintenance analysis. Figure 15 illustrates the messages from Prognostic Maintenance Agent.

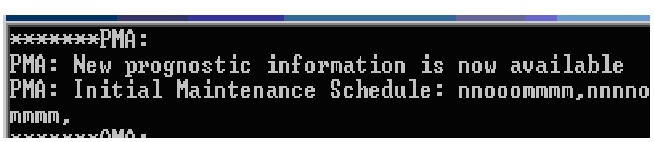


Figure 15: Prognostic Maintenance Agent

Implementation of Opportunistic Maintenance Agent

When the Opportunistic Maintenance Agent (OMA) receives a message sent from PMA, it analyzes different maintenance scheduling alternatives in order to reduce the total cost and increase readiness. The message contains the time frame analysis obtained from individual component evaluation in PMA. OMA analyzes different combinations for a better maintenance schedule by combining some of the maintenance actions at the same time. As seen from Figure 15, PMA classifies the time frames for each component as "No maintenance" shown as "n", "opportunistic maintenance" shown as "o", or "maintenance" shown as "m". OMA comes up with maintenance schedule for this given information. Table 2 illustrates an example of PMA and OMA outputs for five components for nine time frames. In the example, as an output of OMA, component 1, 3, 4, and 5 were scheduled for maintenance at time 3. Component 2 is scheduled for maintenance at time 7. After determination of maintenance schedule, OMA sends message to Resource Planning Agent (RPA) for resource availability check.

Table 2: PMA and OMA Outputs

PMA	Time								
Comp1	n	n	o	o	o	m	m	m	m
Comp2	n	n	n	n	o	o	m	m	m
Comp3	n	n	m	m	m	m	m	m	m
Comp4	o	o	o	o	m	m	m	m	m
Comp5	n	n	o	o	m	m	m	m	m

OMA	Time								
Comp1	n	n	M	n	n	n	n	n	n
Comp2	n	n	n	n	n	n	M	n	n
Comp3	n	n	M	n	n	n	n	n	n
Comp4	n	n	M	n	n	n	n	n	n
Comp5	n	n	M	n	n	n	n	n	n

Figure 16 illustrates the implementation of Opportunistic Maintenance Agent.

Figure 16: Opportunistic Maintenance Agent

Implementation of Resource Management Agent

Resource Planning Agent (RPA) reads the maintenance and inventory database. It receives the maintenance action from MPA or maintenance schedule from OMA and identifies the required resources for the proposed maintenance action(s).

Then, it checks the inventory database to see if the required resources are available. RPA sends an approval or non-approval message to the agent (MPA or OMA).

RPA also performs the resource allocation in the case of limited resources for a given maintenance schedule. Resource allocation is necessary if the resources are not enough for all the recommended maintenance actions but enough for some of them. In this case, RPA identifies which maintenance actions to remove from schedule or to perform by taking minimum risk. In this process, RPA gives priority to maintenance actions that are essential. Figure 17 displays messages from RPA.

Figure 17: Resource Planning Agent

Implementation of Learning Agent

The learning agent learns the three parameters (i.e., RUL threshold, maintenance effectiveness, and resource lead time) by receiving answers to the three questions mentioned above. Figure 18 illustrates the implementation of the learning agent.

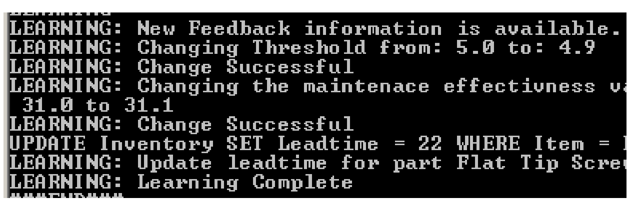


Figure 18: Learning Agent

7. Summary And Future Work

This paper presents a PHM and Maintenance data integration tool that will enable various available diagnostic and prognostic methods to be used in a real environment. This tool provides two methods of interaction: PHM data drives the maintenance actions, and maintenance data creates dynamic learning environment for PHM algorithms. The tool is implemented as intelligent software agents utilizing JADE. The implementation of the tool is also demonstrated in the paper. Future implementation of these techniques will involve migration to USAF airframe maintenance depots and engine overhaul facilities.

Acknowledgement

This work is supported by the Air Force Research Lab/MLLP, Wright Patterson AFB, OH, under SBIR contract No. FA8650-06-M-5229.

REFERENCES

[1] A. Hess and L. Fila, “The Joint Strike Fighter (JSF) PHM concept: Potential impact on aging aircraft problems,” IEEE Aerospace Conference, Mar. 2000.

[2] A. Hess, G. Calvello, and T. Dabney, “PHM a key enabler for the JSF autonomic logistics support concept,” IEEE Aerospace Conference, Mar. 2004.

[3] G. J. Kacprzynski, M. J. Roemer, and A. Hess, “Health management system design: Development, simulation and cost/benefit optimization,” IEEE Aerospace Conference Proceedings, Vol. 6, Page(s):6-3065 - 6-3072. Mar. 2002.

[4] M. A. Schwabacher, “A Survey of Data-Driven Prognostics”, Infotech@Aerospace, paper number: AIAA 2005-7002. Arlington, Virginia. 26 - 29 September 2005.

[5] S. L. Jonathan, D. L. Simon, S. Garg, and et al. “A Survey of Intelligent Control and Health Management Technologies for Aircraft Propulsion Systems,” Journal of Aerospace Computing, Information, and Communication, vol.1 no.12, Pp: 543-563. 2004.

[6] R. Orsagh, D. Brown, M. Roemer, T. Dabney, A. Hess, “Prognostic Health Management for Avionics System Power Supplies,” IEEE Aerospace Conference, Big Sky, Montana. Mar. 2005.

[7] T. J. Wilmering, A. V. Ramesh, “Assessing the Impact of Health Management Approaches on System Total Cost of Ownership,” IEEE Aerospace Conference, Big Sky, Montana. Mar. 2005.

[8] F. Camci, R. B. Chinnam, “Hierarchical HMMs for Autonomous Diagnostics and Prognostics”, Proceedings of the *IEEE World Congress on Computational Intelligence* , Vancouver, British Columbia, Canada, 16-21 July 2006

[9] F. Camci, S. Valentine, “Hierarchical Maintenance Scheduling for Complex Systems Using Mixed Integer Linear Programming”, *Machinery Failure Prevention Technology* ’60, Virginia Beach, VA, 3-6 April 2006

[10] J. R. Bock and T. W. Brotherton and D. Gass, "Ontogenetic Reasoning System for Autonomic Logistics," IEEE Aerospace Conf., Big Sky, Montana. Mar. 2005.

[11] F. Bergenti and et al., “Creating and using ontologies in agent communication”, Proceedings of First International Joint Conference on Autonomous Agents & Multiagents (AAMAS-02), 2002

[12] Y. Zho, et al., “TAGA: trading agent competition in Agentcities”, Proceedings of the 18th International Joint Conference on Artificial Intelligence, 2003

[13] L. Tang, G. J. Kacprzynski, J. R. Bock, “An Intelligent Agent-based Self-evolving Maintenance and Operations Reasoning System” IEEE Aerospace Conf., Big Sky, Montana, Mar. 2006.

BIOGRAPHY

Dr. Fatih Camci is a Senior Project Engineer at Impact Technologies with experience in the development of diagnostic/prognostic strategies for naval ship systems and military aircraft applications. Camci received his PhD in Industrial Engineering from Wayne State University. His M.S. and B.S. degrees are in computer engineering. His expertise includes computational intelligence methods such as neural networks, fuzzy logic, support vector machines and optimization methods including linear, quadratic optimization, and genetic algorithms. He has been involved in development of optimum maintenance scheduling tools and diagnostics/prognostic methods for naval ship systems.

Scott G. Valentine is the Manager, Maintenance & Logistics Systems at Impact Technologies, LLC. His experiences in the RCM and CBM programs extend over a decade. Current efforts support NAVSEA and the Air Force Research Lab in developing decision support tools driving optimal maintenance actions. Mr. Valentine served eleven years active duty in the Navy, qualified in Submarine Officer and Engineer, with his last assignment as Strategic Missile officer on a Trident class submarine. Educational background includes Post Graduate School (MBA) at the University of Rochester; and Undergraduate studies at the United States Naval Academy; Annapolis in Marine Engineering.

Kelly R. Navarra is a Senior Engineer at the Air Force Research Laboratory’s Materials and Manufacturing Directorate. She has a B.S. from the University of Dayton and a M.S. from Virginia Tech, both degrees in mechanical engineering. Ms. Navarra has been with AFRL for 16 years and has over 8 years experience developing diagnostic and prognostics technologies. In her current position she leads the Materials Systems Health Management program activity and serves as the lead program manager on the Enterprise Predictive Analysis Environment (EPAE).